CO456
Web

- most materials adapted from ***Moseley (2007)***, Chapter 4 -

Week 4

CSS for styling ***layout*** and ***navigation***

Module schedule

Wk.	Lecture/subject area(s)	Practical	Reading (Moseley, 2007)
1	Introduction How the Web works	Internet/Web definitions and HTML report	Ch 1 (The way the Web works)
2	HTML 1 (Introductory - inc. lists and hyperlinks)	HTML	Ch 2 pp 24-36 (HTML)
3	HTML 2 (inc. tables, images and forms)	HTML	Ch 2 pp 36-48 (HTML) Ch 3 (XHTML and frames)
4	CSS 1 (Introduction and core CSS principles)	CSS – introductory styles, embedded styles.	Ch 4 pp 76-96.
5	CSS 2 (Positioning elements).	CSS– using IDs, classes and layout control.	Ch 4 pp 97-103.
6	CSS 3 (Advanced layout & navigation)	CSS – using CSS to produce button-like navigation from HTML list elements. (CW2a to be demonstrated).	Specialised articles.
7	JavaScript 1 (Fundamentals, variables)	JS – foundation constructs.	Ch 5 pp 108-116
8	Guided Learning Week	Consolidate Internet & W3 knowledge and HTML & CSS skills.	Review Ch 1 to Ch 4.
9	JavaScript 2 (Functions, branches, loops).	JS – calling functions.	Ch 5 pp 117-124.
10	JavaScript 3 (Objects and the DOM).	JS – manipulating the DOM.	Ch 6 126-139.
11	JavaScript 4 (Forms and validation). And DHTML	JS– validating user completed forms.	Ch 6 139-145, Ch 7.
12	HTML***5***, CSS***3***, - media, forms, gradients, SVG ('Edge') and other enhancements	Web frameworks taster session 1	See practical sheets for information sources
	Vacation		
13	Advanced HTML5, CSS3 & JS frameworks (e.g. jQuery, jQuery Mobile, Box2DWeb)	Web frameworks taster session 2	See practical sheets for information sources
14	Assignment workshop 1	Assignment workshop 1	N/A
15	Assignment workshop 2	Assignment workshop 2	N/A

Positioning with CSS

- The position of HTML elements flows and wraps in the order they are declared in the document.
- CSS provides a means for breaking the normal HTML flow and to position elements out of order and more precisely.
- Using CSS elements may be positioned and shaped as follows
  - ABSOLUTE-ly in a page
  - RELATIVE-ly to the normal flow within a page
  - In ‘depth’ using a z-index
  - By reshaping elements
  - By floating elements “left” or “right”

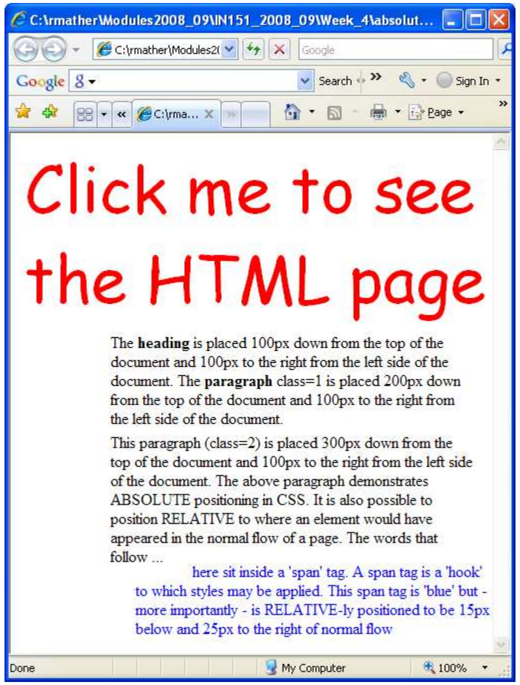
Positioning with CSS

"Absolute" and "Relative"

- Absolute positioning - in the example below
  - the <h1 >heading is placed 100px down and 100px *from* the left side.
  - paragraph 1 <p class="1"> is placed 200px down and 100px from the left.
  - paragraph 2 <p class="2"> is placed 300px down and 100px from the left.

- Relative positioning – in the example below
  - some text is RELATIVE-ly positioned to be 15px below and 25px to the right of normal flow
  - Note – the <span> element is a 'hook' to which styles may be applied.

```
<html><head><style type="text/css">
              h1 {position: absolute; top: 100px; left: 100px}
              p.1  {position: absolute; top: 200px; left: 100px}
              p.2  {position: absolute; top: 300px; left: 100px}
</style></head>
<body>        <h1>An absolutely positioned heading here</h1>
              <p class="1"> … some absolutely positioned text here … </p>
              <p class="2"> … some absolutely positioned text here ...
              <span style="position: relative; top: 15px; left: 25px; color:blue">
              … and … some relatively positioned text here ...</span></p>
</body></html>
```



Click me to see the HTML page

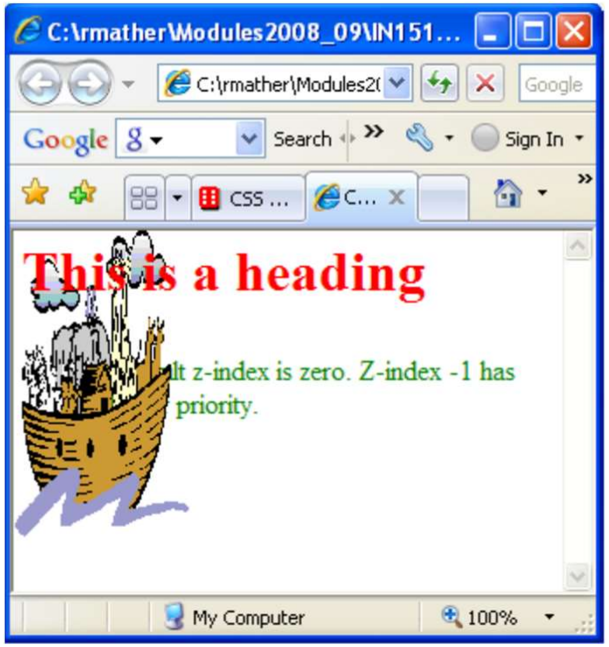
The **heading** is placed 100px down from the top of the document and 100px to the right from the left side of the document. The **paragraph** class=1 is placed 200px down from the top of the document and 100px to the right from the left side of the document.

This paragraph (class=2) is placed 300px down from the top of the document and 100px to the right from the left side of the document. The above paragraph demonstrates ABSOLUTE positioning in CSS. It is also possible to position RELATIVE to where an element would have appeared in the normal flow of a page. The words that follow ... here sit inside a 'span' tag. A span tag is a 'hook' to which styles may be applied. This span tag is 'blue' but - more importantly - is RELATIVE-ly positioned to be 15px below and 25px to the right of normal flow

Positioning with CSS

"z-index"

- The z-index is a number that determines where an object is in a 'vertical' stack of elements
- Higher z numbers place elements closer to the top of the stack
- In this example a paragraph is placed at the bottom (z=0), an image over the paragraph (z=1) and the heading at z=2
- Height and width properties may also be set (see the img.x style in source below)
- Click on the image for the example and source (image courtesy of http://www.horton-szar.net/clipart/boats.php)



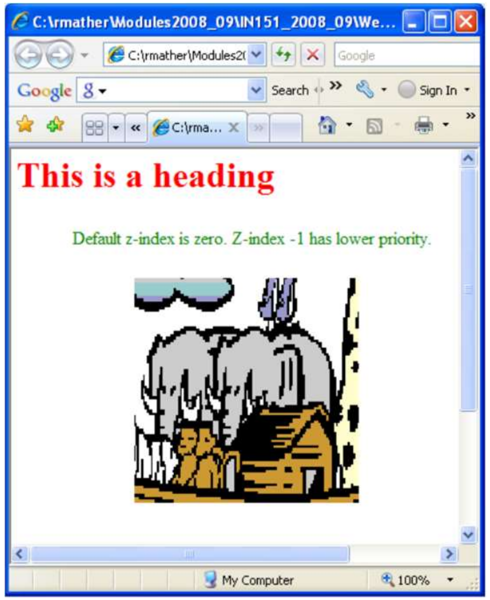
This is a heading

lt z-index is zero. Z-index -1 has priority.

Positioning with CSS

“clip”

- Elements, especially images may be shaped by specifying clip coordinates
- img.x {position: absolute; clip:rect (100px 300px 300px 100px)}
- In this example the elephants are “clipped” from the larger image (*click on image*)
- Another useful property is the “overflow” one which may be set using values
  - “visible” (expands an element so all contents fit - the default value)
  - “hidden” (hides contents that don’t fit inside the element)
  - “scroll” (adds scroll bars so the user may examine the content)



C:\rmather\Modules2008_09\IN151_2008_09\We...

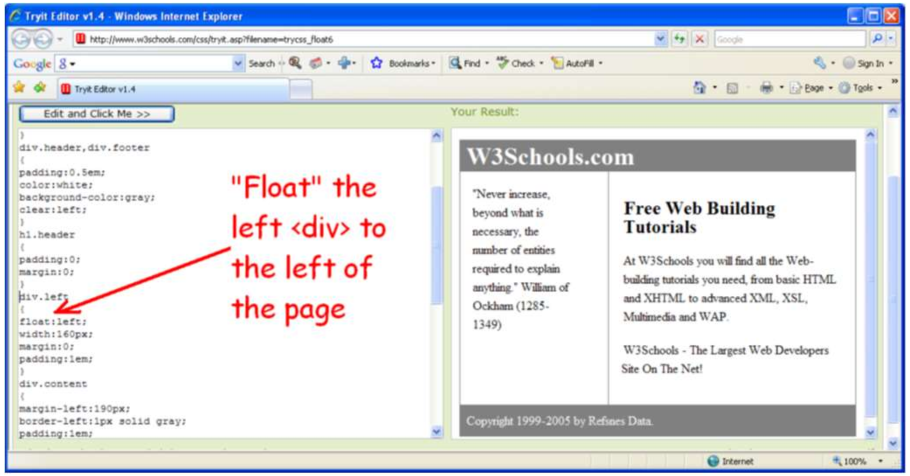
This is a heading

Default z-index is zero. Z-index -1 has lower priority.

Positioning with CSS

“Float”

- Possible to impose relative positioning by floating elements so that other content flows around them
- Float may have values “left” or “right”
- Commonly used for images and for controlling layout with <div> elements
- Problem – implementation of float is inconsistent across browsers
- *Click the image* to visit W3schools example



```
}
div.header,div.footer
{
padding:0.5em;
color:white;
background-color:gray;
clear:left;
}
h1.header
{
padding:0;
margin:0;
}
div.left
{
float:left;
width:160px;
margin:0;
padding:1em;
}
div.content
{
margin-left:190px;
border-left:1px solid gray;
padding:1em;
```

"Float" the left <div> to the left of the page

W3Schools.com

"Never increase, beyond what is necessary, the number of entities required to explain anything." William of Ockham (1285-1349)

Free Web Building Tutorials

At W3Schools you will find all the Web-building tutorials you need, from basic HTML and XHTML to advanced XML, XSL, Multimedia and WAP.

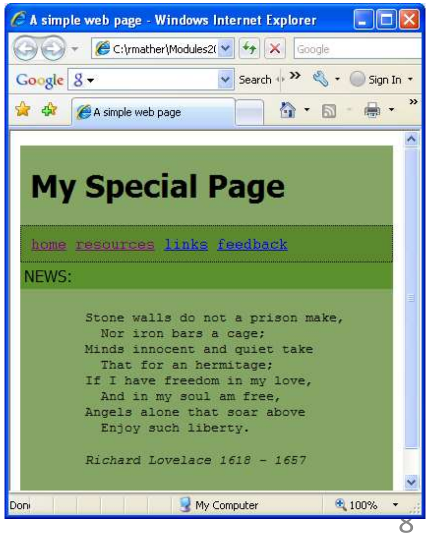
W3Schools - The Largest Web Developers Site On The Net!

Copyright 1999-2005 by Refsnes Data.

Positioning with CSS

“<div>”

- The intention of CSS => separate formatting and styling rules from content
- Use of CSS and <div> (division) elements to independently position and style logical sections of pages is a ***very powerful way to impose structure and layout***
- Commonly used for images and for controlling layout with <div> elements
- Generally base layouts on identifiers e.g.
  - CSS = #navigation{padding: 10px; border: 1px … background-color:#5c892e }
  - Implementation in <body> = <div id="navigation">
- Problem – implementation of float is inconsistent across browsers
- *Click the image* to visit the example



My Special Page

home resources links feedback

NEWS:

```
Stone walls do not a prison make,
  Nor iron bars a cage;
Minds innocent and quiet take
  That for an hermitage;
If I have freedom in my love,
  And in my soul am free,
Angels alone that soar above
  Enjoy such liberty.

Richard Lovelace 1618 - 1657
```

Positioning with CSS

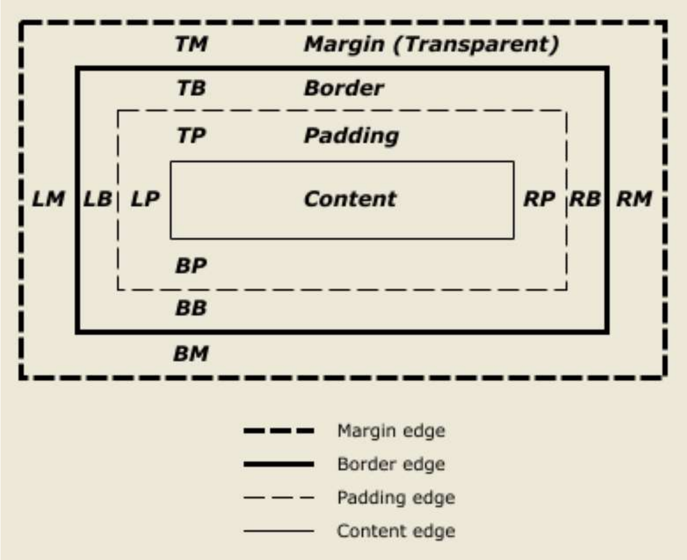
Advanced Layouts and Navigation

- Common practice is to nest all <div> elements inside a single “container”
- Examples of nested elements may include separate <div> identifiers for “head”, “navigation”, “content”, “foot” panels.
- Multiple column layouts may have additional identifiers, for example a three column layout may have “navigation”, “contentMain” and “contentRight”
- Centre, float and height positioning of <divs> is problematic due to cross-browser incompatibilities
- Many developers adopt special XML HTML “Quirks mode” to ensure backward compatibility across browsers e.g.
  - <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">
  - See comprehensive reference on http://en.wikipedia.org/wiki/Quirks_mode
- … see later for an example …

The Box Model

(materials provided by W3C and W3Schools)

"Each box has a *content area* (e.g., text, an image, etc.) and optional surrounding *padding*, *border*, and *margin* areas; the size of each area is specified by properties defined below. The following diagram shows how these areas relate and the terminology used to refer to pieces of margin, border, and padding ... " W3C (http://www.w3.org/TR/CSS2/box.html)

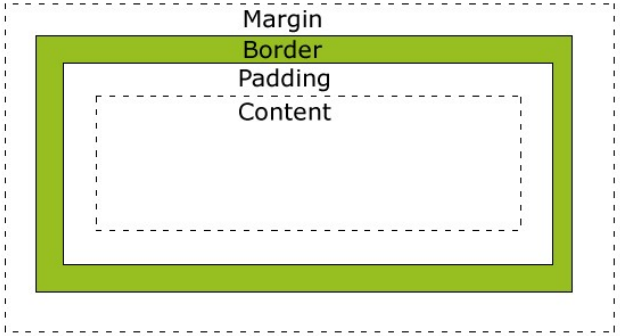


Suggest that for a better understanding of the box model visit W3Schools and read ...

http://www.w3schools.com/css/css_boxmodel.asp

... Then "try it" using their interactive frames here ...

http://www.w3schools.com/css/tryit.asp?filename=trycss_boxmodel_width



Explanation of the different parts:

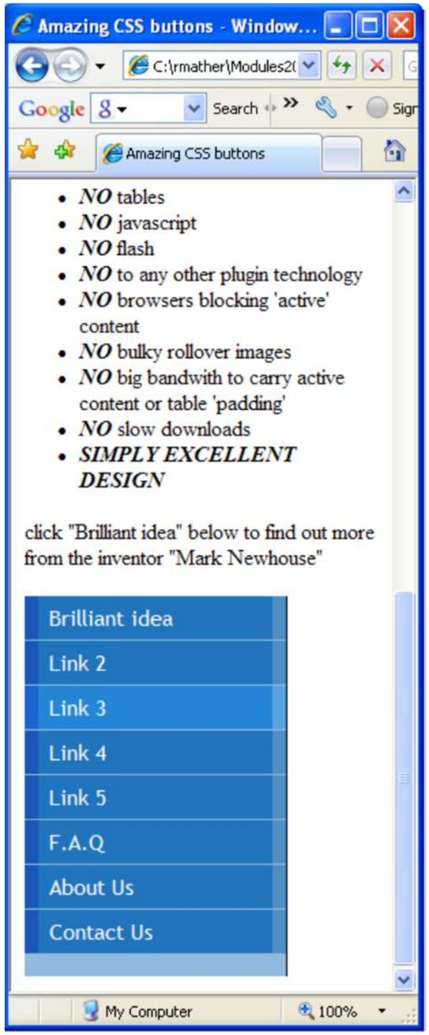
- **Margin** - Clears an area around the border. The margin does not have a background color, it is completely transparent
- **Border** - A border that goes around the padding and content. The border is affected by the background color of the box
- **Padding** - Clears an area around the content. The padding is affected by the background color of the box
- **Content** - The content of the box, where text and images appear

Positioning with CSS

Advanced Layouts and Navigation

- Aim – “imageless” 3D button like navigation w/o bandwidth overhead
- Solution – the humble HTML list – can modify borders, colours, rollover effects so that they have all the appearance of a button
- One example here … do examine Mark Newhouse’s ‘buttons’!
- HTML could be as follows – CSS does the rest!

```
<div id=“NavigationButtons">
        <ul>
                    <li><a href="http://www.MySite/home"</a></li>
                    <li><a href="http://www.MySite/page1">Page 1</a></li>
                    <li><a href="http://www.MySite/page2">Page 2</a></li>
                    <li><a href="http://www.MySite/page3">Page 3</a></li>
        </ul>
</div>
```



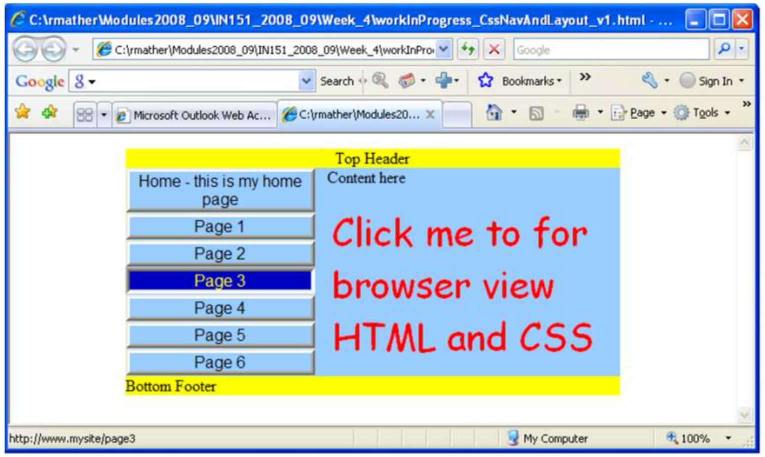
- ***NO*** tables
- ***NO*** javascript
- ***NO*** flash
- ***NO*** to any other plugin technology
- ***NO*** browsers blocking 'active' content
- ***NO*** bulky rollover images
- ***NO*** big bandwith to carry active content or table 'padding'
- ***NO*** slow downloads
- ***SIMPLY EXCELLENT DESIGN***

click "Brilliant idea" below to find out more from the inventor "Mark Newhouse"

Positioning with CSS

Putting it all together - Advanced Layouts ***and*** Navigation

- Possible to produce easily modifiable templates for layout and navigation
- This example combines a 2 column header and footer layout with 3D CSS button navigation.
- The layout is uses CSS identifiers to control div position
- The main "container" <div> is set to 70% and centred by setting margin-left and margin-right properties to be equal using the 'auto' value.
- 3D rollover button effects use CSS border: 4px inset/outset properties
- Investigate/modify the mechanism by examining the HTML and CSS
- For more on float problems and the need to clear floats go to … http://css-tricks.com/all-about-floats/


C:\rmather\Modules2008_09\IN151_2008_09\Week_4\workInProgress_CssNavAndLayout_v1.html - …
C:\rmather\Modules2008_09\IN151_2008_09\Week_4\workInPro
Google
Google
Search
Bookmarks
Sign In
Microsoft Outlook Web Ac…
C:\rmather\Modules20…
Page
Tools
http://www.mysite/page3
My Computer
100%

Top Header

Home - this is my home page

Page 1

Page 2

Page 3

Page 4

Page 5

Page 6

Content here

Click me to for browser view HTML and CSS

Bottom Footer

References and sources

- Create CSS buttons quickly here http://www.thesitewizard.com/wizards/css-menu-buttons.shtml
- Learn about float problems and the need to “clear” floats http://css-tricks.com/all-about-floats/
- Online book covering many aspects of design http://www.webstyleguide.com/
- Mark Newhouse’s solutions to CSS navigation http://alistapart.com/article/taminglists
- Ross Shannon on many CSS subjects including changing the look of browser scroll bars http://www.yourhtmlsource.com/stylesheets
- Wikipedia on CSS http://en.wikipedia.org/wiki/Cascading_Style_Sheets
- Nick Rigby on CSS dropdown menus http://www.alistapart.com/articles/horizdropdowns
- Matthew James Taylor’s layouts and navigation http://matthewjamestaylor.com/blog/perfect-multi-column-liquid-layouts
- W3C The Box Model http://www.w3.org/TR/CSS2/box.html
- W3Schools The Box Model http://www.w3schools.com/css/tryit.asp?filename=trycss_boxmodel_width